Technical Disclosure Commons

Defensive Publications Series

December 28, 2018

Tool For Measuring End To End Latency

Michal Kielan

Jose M. Gonzalez

Michal Godek

TOOL FOR MEASURING END TO END LATENCY

ABSTRACT

A system, method, and computer readable media for measuring end to end (e2e) latency in a video calling system is disclosed. The latency measuring system includes a video calling system having at least two user devices that may be connected over a network, one or more external cameras, a device producing luminescence and a computing device. The camera is placed in front of each of the user devices and is configured to record the video call in both the devices. The luminescence device may be turned on for a short moment during the recording of the video call. The system is configured to capture an original time, a self-view time and a remote time and to calculate the e2e and the e2s latency to support matching each frame. The method allows evolution of the e2e and e2s latencies on a per-frame basis, thereby providing better statistics of video call quality.

KEYWORDS: video calling, latency measuring system, frame rate, image processing, QR code, end to end latency, e2e, self-view latency, e2s.

BACKGROUND

Video calls augment voice calls with live visual interaction between users. Users prefer to make untethered video calls, often using their mobile devices such as smartphones, tablets and laptops or dedicated video calling devices, instead of sitting in front of their desktop computers. Mobile devices are connected to the Internet through WiFi or cellular networks. Video call applications such as FaceTime, Hangouts, WeChat and Skype are popular on various mobile platforms. Due to the higher bandwidth requirement of video calls, it is much more challenging to deliver high-quality video calls than voice calls. During voice calls over a network, long delays between transmission and reception of signals are easily noticeable and off-putting.

One of the common causes of poor quality of voice or video communication over a network is high or excessive latency. High latency is caused as a result of lag in hardware, software, and network that are part of the video calling system. High latency causes communication breakdown and stops conversation flow. Users want latency to be as low as possible for various applications such as operating remote devices, streaming live events, video conferencing, etc. For example, during a video conference session, it is desirable to have real time conversations without delays between speaking and hearing of responses. Low latency for video calls is usually defined as less than 100 milliseconds. Latency effects depend on the observer, but most will perceive obvious latency around 100 - 120 milliseconds. Communications will start to break down around 250 – 300 milliseconds.

Several methods have been described for testing if a video system's latency meets the user requirements. However, many of these approaches are limited by their ability to measure multiple aspects that contribute to latency. Further, such methods are not readily applicable for all systems. There is a need to measure the lag developed in the hardware, software, and network that contribute to the latency between two devices during a call.

DESCRIPTION

A system, method, and computer readable media for measuring latency in a video calling system is disclosed. As defined herein, latency includes any lag due to hardware, software and network from the moment an event happens in the sender side of the call, to the moment the receiver sees and hears that event. The system is configured to measure end to end (e2e) latency between at least two paired computing devices connected through a network and a self-view latency (e2s) for each of the connected devices. As defined herein, the e2e latency is the time delay between a photon touching the screen of the sender device and its visibility in a remote receiver device. As defined herein, the e2s latency is the time delay between a photon touching the screen of the sender device and its visibility in a self-view window of a video calling application.

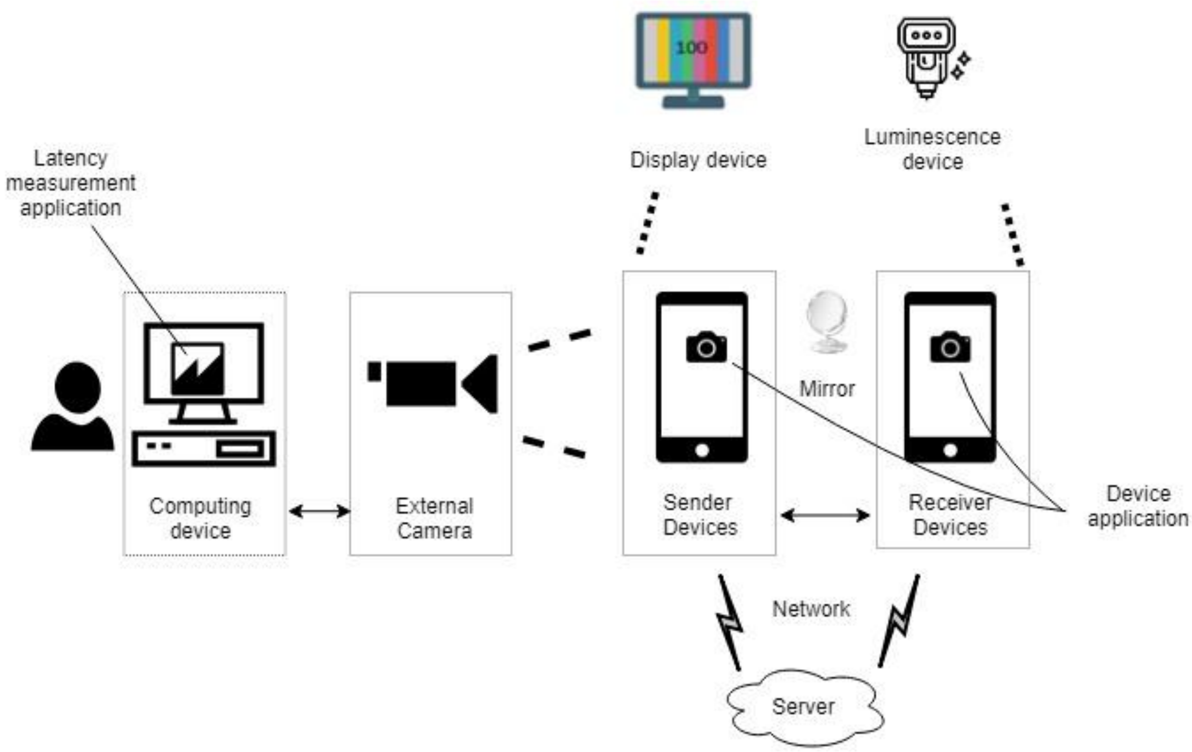

FIG. 1A: System to measure latency in a video calling system

The latency measurement system, as illustrated in FIG. 1A, may include a video calling system having at least two user devices that may be connected over a network. In a typical scenario, multiple user devices may be able to send (broadcast) and receive (view) video streams via the video calling system. The streaming may be peer to peer and/or via a server. The method includes running a device application configured to display an interface for accessing the built-in camera functionalities. The device application may be installed in the device or an application actuated via a hyperlink on a browser page. The application may stream video from one or more user devices connected to other devices over the network. The connected user devices may have interfaces that support streams from both self and other connected devices to be displayed. The installed application may include access to a list of contacts such as social networking contacts. The devices may be configured for achieving optimal latency during video calls through improvements to one or more of hardware, processing speed, transmission speed, and video quality. The video calling system may include encoders, decoders, means to encapsulate the encoded data as packets, a transmission protocol to transmit the packets through a network, and a play-out buffer to direct packets in the right order and to buffer enough data for the decoder. The network connecting the sender and the receiver device may have a predetermined bandwidth and may include one or more switches, routers, proxies and firewalls.

The latency measurement system, as illustrated in FIG. 1A, further includes one or more of external cameras, a luminous object or a device producing luminescence, a display with frame counter, a mirror arrangement and a computing device. The computing device may include a processing unit, a memory unit coupled to the processing unit including an application for measuring latency and various modules such as an image processing module, a calculation module and a display module. The external camera, luminescence object and computing device may be manually operated or may execute instructions automatically. For instance, the one or more external cameras may be interfaced to the computing device for automatically capturing and transmitting captured media. The external camera and luminescence object may be part of a single device such as a mobile phone camera with a built-in flash.

The sender device and receiver device may be positioned next to each other. The external camera may have a frame rate of greater than or equal to 240 fps. The external camera is placed in front of the user devices and is configured to record the video call in both the devices. The luminous object or the device producing luminescence may produce glare and is turned on during the recording of the video call. The luminescence device produces a glare on the screen of the sender device and the receiver device. At least an original time, a self-view time and a remote time are captured. The original time includes the moment at which a glare is produced by the luminescence device on the display screen of the devices. The self-view time includes the moment at which the luminescence appears in the self-view window of each of the devices. The remote time includes the moment at which the luminescence appears in the remote display of the paired devices. The frames from the recordings are received by the computing device for calculating the time difference between the frames. The differences and hot spots are detected as inputs for the image processing module. The e2e latency and the e2s latency are calculated from the captured original time, the self-view time and the remote time. In some aspects, the measurement granularity is the inverse of the camera fps. The system may automatically calculate the e2e latency and the e2s latency by knowing the frame in which the luminescence occurred, and how many flashes were expected in the recorded video.

The luminous object or device is a flashlight as shown in FIG. 1B. The flash light may be turned on for a short moment during the recording of the video call to produce a glare on the screen of the sender device. The system captures one or more moments from the video. At a first instance the system captures the original time that is the moment at which a glare is produced by the flashlight on the display screen of the sender device. The self-view time, the moment at which the flashlight appears in the self-view window of the sender device is captured at a second instance and at a third instance the remote time, the moment at which the flashlight appears in the remote display is captured. The e2e latency and the e2s latency are obtained from the captured original time, the self-view time and the remote time.

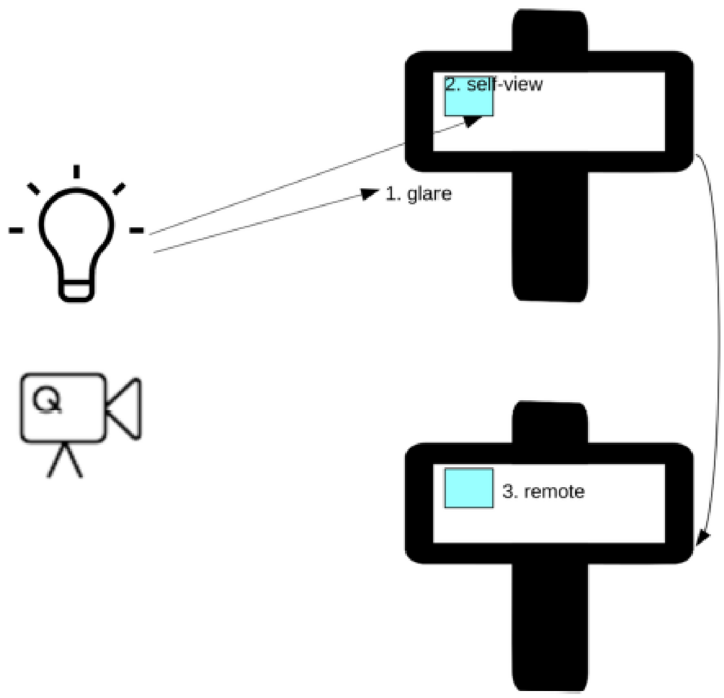

FIG. 1B: An exemplary system for measuring the latency in a video calling system

The video call session is initially recorded with the external camera placed in front of the user devices, as shown in FIG. 1B. The flash is turned on for a short moment while recording the video call. The flash creates a glare on the screen of the user devices at a first instance, the luminance is visible in the self-view of the sender device at a second instance and the luminance is visible in the remote display i.e., in the display of the receiver device at a third instance. The method includes calculating the e2e latency by measuring the time between the first instance and the third instance. The method further includes calculating the e2s latency by measuring the time between the first instance and the second instance. The method may be performed by obtaining one sample per second. To automate the process the image processing module includes an image processing algorithm. The image processing algorithm receives frames from the videos, and converts the RGB images to black and white. The algorithm applies a filter that obtains a threshold value for the frame. The number of white pixels in the frame that have a value above the threshold value is calculated. The white pixels are further compared against the white pixels in the previous frame to identify the difference in the luminance of the two frames. When the difference between two frames is reasonably high, it means that the luminous object or the device producing luminescence is switched from on to off, or from off to on. The latency may be calculated by identifying the frame in which the flash was turned on, and how many flashes were recorded in the video.

Alternatively, the system may include a video display having a frame counter as shown in FIG. 1C. A mirror is placed between the two user devices. The user devices are pointed towards the video display. The frame counter may include human-readable and/or machine-readable modules such as a quick response (QR) code. The frame counter is configured to read each frame of the video recording to extract the original time, the self-view time and the remote time. The method may also be extended to an alternate system as shown in FIG. 1D. In the alternate system, the method is implemented by matching each frame of a video of the video calling system. The method may include obtaining one sample per frame to monitor the evolution of the e2e and the e2s latencies on a per-frame basis. The video call is recorded with the camera placed in front of the user devices. A video is displayed in a video display having a frame counter. The displayed video may be seen in the mirror, the sender device and the receiver device. The method further includes passing each frame of the video through a QR code reader. The original time, self-view time and the remote display time is obtained from the QR code reader and the e2e and the e2s latencies may be obtained on a per-frame basis.

In one example, a video call was started between two user devices. The system was allowed to stabilize for two minutes and the video call was recorded using a high frame rate camera. A torch was turned on for 1 to 2 seconds and was turned off thereafter. The recording was stopped and the video was analyzed. Exemplary images obtained from the camera recordings are shown in FIG. 2A, FIG. 2B and FIG. 2C. FIG. 2A shows the instant at which the torch was switched on. The light from the torch can be seen as a glare in the screen as shown in FIG. 2A. The flash light was noticed in the self-view window at a later instance as shown in FIG. 2B. The flash light in the remote window is as shown in FIG. 2C. The e2e and the e2s latencies were calculated to be 112.5 ms and 408.333 ms, respectively.

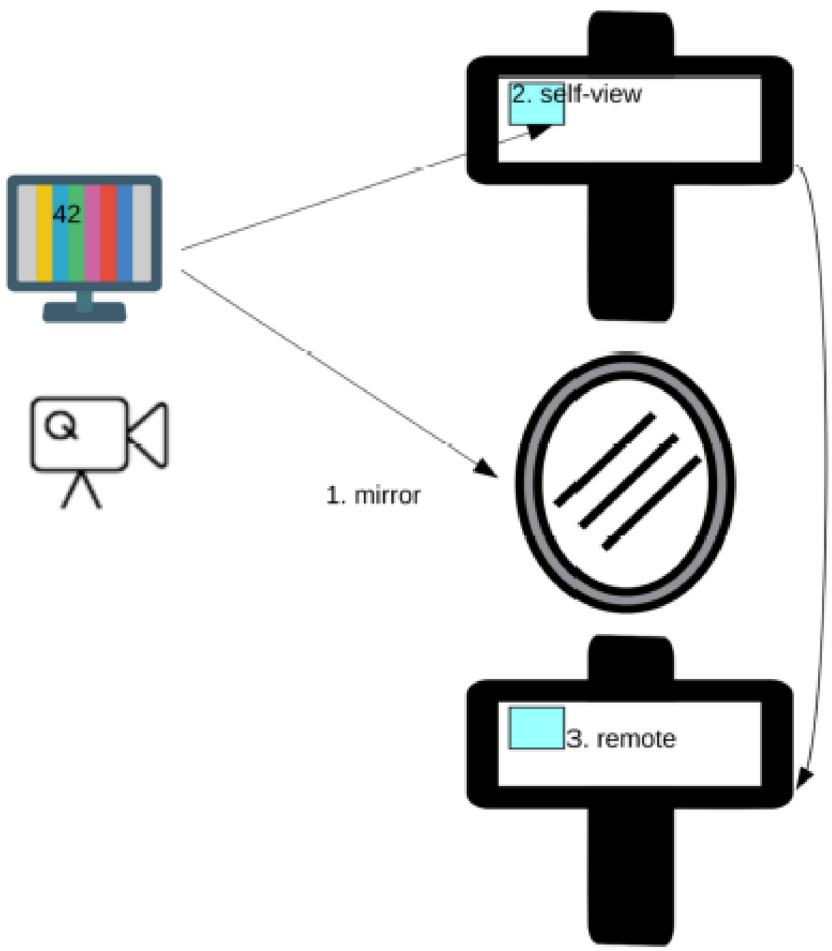

FIG. 1C: An alternate exemplary system for measuring the latency in a video calling system

FIG. 2A: The flash light appears as a glare in the screen

FIG. 2B: The flash light is seen in the self-view window. The right device has swapped the self-view and the remote window

FIG. 2C: The flash light is seen in the remote window

In another example, an image processing algorithm that is implemented in a processor was used to automate the process. A picture as shown in FIG. 3A was taken using a mobile device with the flash switched on for a time duration. The image processing algorithm received the frame and converted the RGB frame to black and white as shown in FIG. 3B. A threshold value for the frame was obtained by using the filter in the algorithm. The number of white pixels in the frame as shown in FIG. 3C that had a value above the threshold value was calculated. The white pixels were further compared against the white pixels in the previous frame to identify the difference in the luminance of the two frames. When the difference between two frames was above the threshold, the algorithm identified the instance as the moment at which the flash was switched from on to off, or from off to on. The difference in white pixels was calculated as

$$diffPixels = frameWhitePixels - lastFrameWhitePixels \ldots\ldots\ldots (1),$$

where $diffPixels$ is the difference in the white pixels in the current frame and the white pixels in the previous frame, $frameWhitePixels$ is the white pixels in the current frame and $lastFrameWhitePixels$ is the white pixels in the previous frame. FIG. 3E shows an example of the difference in luminance in each frame of the recorded video. From the graph it is found that the flash was triggered in frame 30(original), 56(self-view) and 125(remote-view). This corresponds to a total latency of 100 frames.

FIG. 3A

FIG. 3B

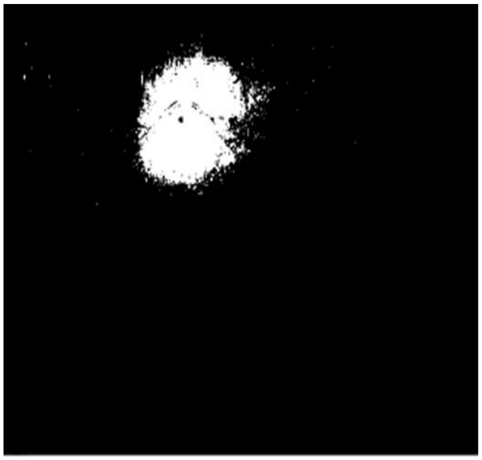

FIG. 3C

FIG. 3D

FIG. 3A: Capturing an image. FIG. 3B: Convert the RGB to black and white. FIG. 3C: White pixels in the current frame. FIG. 3D: Difference in white pixels in the current frame and the previous frame

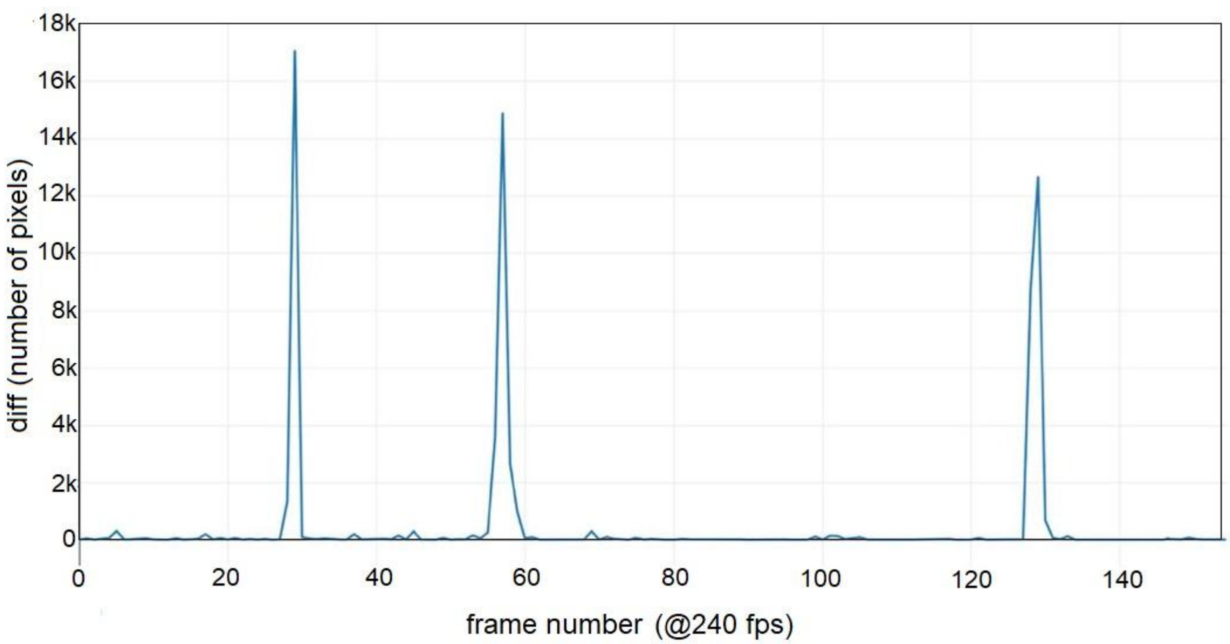

FIG. 3E: An example of the difference in luminance in each frame of the recorded video

In yet another example, a mirror was kept between two user devices that were connected over a network. The display of the two user devices and the mirror were pointed towards a video display that had a frame counter. The video in the video display was started. The video was visible in all the three screens. Each frame of the video was further passed through a QR code reader. Three values each were obtained for three timestamps 100ms, 124ms and 130ms as shown in FIG. 4A. At time 130ms, the QR code reader returned 3 numbers, namely 130, 124, and 100. The number 130 corresponds to the frame that was seen in the mirror at the actual timestamp 130. The number 124 is the frame that was seen in the self-view. This means that the self-view is 6 frames after the frame in the mirror. Therefore the e2s latency is 100 ms or 6 frames at 60 fps. The number 100 is the frame that is seen in the remote video. This means that the frame in the remote view is 30 frames after the mirror, or 500 ms or 30 frames at 60 fps. FIG. 4B shows a 4x5 matrix of 20 continuous samples taken using a 240 fps camera in front of a machine with a monitor playing a 120 fps video file and each frame adds 1 to the counter. It is noted that the monitor is actually displaying a 60 fps video and that half of the frames are being dropped somewhere, and each frame is being captured 4 times.

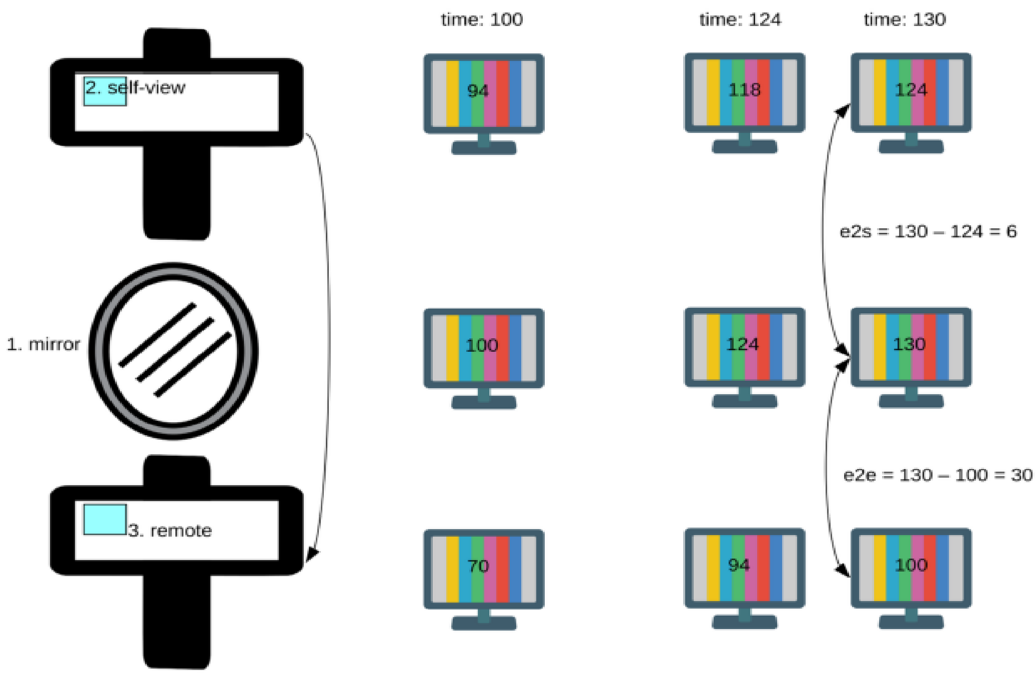

FIG. 4A: An example of a frame obtained using the alternate system

FIG. 4B: A 4×5 matrix of 20 continuous samples taken using a 240 fps camera

The disclosed system, method and computer executable media allow for each frame to be matched to allow taking one sample per frame and monitor the evolution of the e2e and e2s latencies on a per-frame basis. This will result in both efficient and improved latency measurements attributed to hardware, software, and network in video calling systems, thereby providing accurate measure of video quality.